

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

ZÁKLADNÍ KONFIGURÁTOR SYSTÉMU INELS PRO KONCOVÉ ZÁKAZNÍKY

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

PAVEL KOČÍŘ

BRNO 2010



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

ZÁKLADNÍ KONFIGURÁTOR SYSTÉMU INELS PRO KONCOVÉ ZÁKAZNÍKY

BASIC CONFIGURATOR OF INELS SYSTEMS FOR END CUSTOMERS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Pavel Kočíř

VEDOUCÍ PRÁCE

SUPERVISOR

Doc. Ing., Dipl.-Ing. Martin Drahanský
Ph.D.

BRNO 2010

Abstrakt

Tématem této bakalářské práce bylo navrhnout webovou aplikaci pro konfiguraci systému inteligentní elektroinstalace INELS od firmy ELKO EP s.r.o.

První část vysvětluje pojem inteligentní elektroinstalace a systém INELS. Dále také analyzuje možnosti programovacích jazyků, které používám při vývoji aplikace.

Druhá část popisuje návrh, realizaci a výsledky testování aplikace pro konfiguraci.

Klíčová slova

inteligentní dům, inteligentní elektroinstalace, webový konfigurační systém

Abstract

The theme of this bachelor thesis was to design web-application for configuration for the system of intelligent electrical INELS from company ELKO EP s.r.o.

The first section explains the concept of intelligent wiring systems and the system INELS. Further Also it analyze the possibilities of programming languages that I use when developing the applications.

The second part describes the design, implementation and results of testing the application for configuration.

Keywords

Intelligent house, intelligent electrical installation, web-based configuration system

Citace

Kočříř Pavel: Základní konfigurační systém INELS pro koncové zákazníky, bakalářská práce, Brno, FIT VUT v Brně, 2010.

Základní konfigurator systému INELS pro koncové zákazníky

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením Doc. Ing., Dipl.-Ing. Martinem Drahanským Ph.D. Další informace mi poskytla firma ELKO EP s.r.o., jmenovitě Jiří Konečný a Veronika Dýšková.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Pavel Kočíř
15.5.2010

Poděkování

Chtěl bych poděkovat panu Doc. Ing., Dipl.-Ing. Martinu Drahanskému Ph.D. za odborné a ochotné konzultování mé bakalářské práce a užitečné rady. Dále bych chtěl poděkovat firmě ELKO EP s.r.o., jmenovitě Jiřímu Konečnému jako jednatele firmy za administrativní záležitosti a odborné rady a taktéž Veronice Dýškové za ochotné vyřizování mých žádostí s firmou.

© Pavel Kočíř, 2010.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů..

Obsah

Obsah	1
1 Úvod.....	3
2 Inteligentní elektroinstalace	4
2.1 Pojem „Inteligentní elektroinstalace“	4
2.2 Historie	4
2.3 Dnešní doba.....	5
2.3.1 Systém INELS	5
3 Aktuální stav dané problematiky	8
3.1 Problematika uživatelského rozhraní	8
3.1.1 Teorie	8
3.2 Jazyk HTML	9
3.2.1 Teorie	9
3.2.2 Standardizační organizace	10
3.2.3 Struktura dokumentu HTML	11
3.3 Jazyk SQL	12
3.3.1 Teorie	12
3.3.2 Historie.....	13
3.3.3 Příkazy jazyka SQL	14
3.4 Jazyk PHP	15
3.4.1 Teoretický rozbor.....	15
3.4.2 Historie.....	16
3.4.3 Vkládání PHP do HTML	16
3.4.4 Popis struktury PHP.....	17
3.5 JavaScript	17
3.5.1 Teorie	17
3.6 Kaskádové styly CSS	18
3.6.1 Popis.....	18
3.6.2 Syntaxe jazyka	19
4 Návrh aplikace	21
4.1 Obsah aplikace	21
4.2 Vzhled aplikace	21
4.3 Funkce aplikace.....	22
5 Implementace aplikace.....	23
5.1 Požadavky aplikace	23

5.2	Vývoj aplikace.....	24
5.2.1	Vývoj uživatelského rozhraní aplikace	24
5.3	Rozdělení aplikace	26
5.3.1	Administrátorské rozhraní	26
5.3.2	Konfigurátor.....	28
6	Testování.....	33
7	Závěr	34
	Literatura	35
	Seznam příloh	37

1 Úvod

Inteligentní elektroinstalace je pojem, který je v dnešní době vysvětlován v mnoha různých významech. V širším smyslu se jedná o budovu, která je vybavena komunikační a počítačovou technikou. Tato vybavenost předvídá a reaguje na potřeby uživatelů budov s cílem zvýšit jejich pohodlí a komfort, poskytnout bezpečí a snížit náklady na provoz budovy v podobě úspory energií. Velmi často můžeme slyšet slova jako „chytrý dům“, „domácí automatizace“, „inteligentní dům“ nebo „domotika“.

Tato technika se již stále více uplatňuje i v rodinných domech a není to již jen výhradou nových moderních center. Jedním ze systémů ve sféře inteligentní elektroinstalace je i systém INELS firmy ELKO EP s.r.o. (www.elkoep.cz, www.inels.cz) jako jeden z mnoha zástupců v této oblasti.

První myšlenky a koncepce na tyto inteligentní domy existují již bezmála 50 let. Jednalo se především o koncepty automatizovaných domů ve kterých byli různí roboti na úklid podlah, v jednotlivých pokojích bylo možno přehrávat audio/video nahrávky apod.

Inteligentní elektroinstalace jistě v budoucnu přinesou razantní změnu způsobu bydlení oproti tomu dnešnímu. Především ve změně vztahu lidí a techniky a také ve změně použití technologií v domácnostech. I kvůli těmto důvodům jsem si vybral téma bakalářské práce z této rozsáhlé oblasti inteligentních elektroinstalací.

Aby bylo možné efektivně, rychle a bez zbytečných nákladů systém inteligentní elektroinstalace zakomponovat do budov, je nezbytné si jen nejprve vhodně namyslet a navrhnout před následnou instalací. Pro asi nejjednodušší a nejpohodlnější návrhy slouží počítače a různé konfigurační programy, které se na těchto počítačích dají řešit. A právě jeden takový konfigurační systém je námětem této bakalářské práce. Bude se přesně jednat o webový konfigurační systém. Tento systém by měl dále sloužit pro různé návrhy systému INELS pro koncové uživatele.

Tato bakalářská práce popisuje jak návrh tohoto systému, tak i jeho následnou realizaci. Práce byla zadána firmou ELKO EP s.r.o. (www.elkoep.cz) a je rozdělena do několika kapitol.

V kapitole 2 lze nalézt popis a vysvětlení pojmu „inteligentní elektroinstalace“. Je zde stručně vysvětlena její historie, vyjmenováno pár firem, které se touto problematikou a instalací zabývají a rozbor systému INELS. V kapitole 3 nalezneme aktuální stav dané problematiky a teorie, které se dnes využívají pro tvorbu vhodných konfiguračních systémů. Jsou zde popsány různé jazyky, které byly k vytvoření tohoto konfiguračního webového nástroje zapotřebí. Kapitola 4 popisuje samotný návrh celého konfiguračního systému včetně požadavků na tento systém.

2 Inteligentní elektroinstalace

Tato kapitola popisuje a rozebírá pojem „Inteligentní elektroinstalace“, její stručnou historii a nastínění využití této technologie v reálném světě. Je zde popsán systém INELS, co tento systém je, jeho historie, funkčnost a porovnání s jinými systémy. Pro zpracování této kapitoly bylo využito zdrojů [1, 2, 3, 4, 5, 6, 7, 8, 9].

2.1 Pojem „Inteligentní elektroinstalace“

Citace: „**Intelligence** (z lat. *inter-legere*, rozlišovat, poznávat, chápat) je rozumová schopnost řešit nově vzniklé nebo obtížné situace; schopnost učit se ze zkušeností; schopnost přizpůsobit se; schopnost správného určení podstatných souvislostí a vztahů, pomocí nichž řešíme nové problémy a orientujeme se v nastalých situacích.“[1] Tato citace je sice brána z pohledu člověka, jako myslící osoby, ale i tak velmi dobře vystihuje, jak můžeme chápat pojem inteligence i v oblasti bydlení.

Citace: „**Elektroinstalace** je soustava elektrotechnických zařízení k vedení a ovládání elektrického proudu nebo elektrických signálů v místě jejich užívání. Elektroinstalace slouží k přenosu elektrické energie (silové rozvody), nebo k přenosu dat a slaboproudých signálů.“[2] Jedná se tedy o soustavu vzájemně propojených vodičů, ochranných prvků (jističe, pojistky aj.) a spínacích zařízení (např. spínače a relé).

Inteligentní elektroinstalace je tedy systém pro řízení budov. Někdy můžeme i zaslechnout pojem „chytrý dům“ nebo jiný podobný název. Umožňuje vypínání a zapínání stojů a přístrojů, stejně tak i jejich hlídání, kontrolu a následnou regulaci. Tento systém rovněž zvládá i kontrolu zabezpečení domů a jejich hlídání. Všechny tyto události se dějí pomocí signálů, které vysílají snímače, které jsou rozmístěné na strategických bodech v domě. Tyto signály o změnách stavu se zasílají buď do centrální jednotky (centralizovaný systém), která je vyhodnotí a regulátorům zašle patřičné povely, nebo se signály posílají přímo jednotlivým zařízením v případě decentralizovaného systému (v každém zařízení se nachází vlastní mikroprocesor pro zpracování dat).

2.2 Historie

Rok 1897 lze považovat za počátek zrodu inteligentní elektroinstalace. V tomto roce byla založena společnost Instabus Gemeinschaft, která vznikla z firem Berker, Merten, Gira a Siemens. Cílem bylo vyvinout systém, který by uměl měřit, regulovat, řídit a sledovat stavy v budovách. Po obrovském zájmu evropských výrobců elektroinstalační techniky se z Instabus Gemeinschaft stala nadnárodní nezávislá organizace.

V roce 1990, přesně 8. května, vznikla tedy asociace EIBA (European Installation Bus Association) se sídlem v Bruselu. Jejím cílem je nabízet jednotnou a kvalitní technologii. Vzniklo logo EIB jako ukazatel této kvality. Pro standart EIB byla vytvořena i norma, je začleněna do soustavy CELENEC. Hlavními přínosy této standardizace je bezchybná spolupráce jednotlivých zařízení od několika různých výrobců v jedné budově.

2.3 Dnešní doba

V dnešní době se pojem inteligentní elektroinstalace stává stále používanějším. Náklady na provoz budou neustále rostou a lidé chtějí ušetřit všude, kde se jen dá. Na dnešní trhu můžeme nalézt spoustu různých systémů inteligentní elektroinstalace od různých firem. Příkladem mohou být systémy EGO-N, INELIN, Astrom plus nebo INELS. Právě INELS je systém, který nás momentálně bude zajímat nejvíce.

2.3.1 Systém INELS

Systém INELS je výrobkem firmy ELKO EP s.r.o. (www.elkoep.cz).

2.3.1.1 Co tento systém znamená a zvládá

Systém INELS je systémem inteligentní elektroinstalace. Řídí provoz budov od ovládání osvětlení a rolet, přes klimatizaci a regulaci vytápění až po zabezpečení majetku. Zabezpečení systému je jak z pohledu prevence vzniku požáru nebo úniku plynu, tak i z pohledu vloupání do objektu. V delší nepřítomnosti v budově, systém tuto nepřítomnost dokáže nahradit např. náhodným rozsvěcováním jednotlivých světel v různých částech budov, čímž simuluje přítomnost člověka v objektu. Rozpoznává změny prostředí oproti požadovanému stavu a informace o těchto změnách poskytuje prostřednictvím internetu, mobilního telefonu nebo PDA. Jeho návrh dokáže uspokojit jak malé elektroinstalace, tak i rozsáhlé celky, které vyžadují různé automatizace a úplnosti.

Tento systém je možné využít v rodinných domech nebo bytech. Stejně dobře se hodí pro různé administrativní budovy a prodejní prostory. Nemalé zastoupení může mít i v oblastech rozsáhlých budov, různých komplexů budov i v průmyslové sféře.

2.3.1.2 Jak funguje

Tento systém je centralizovaný.

Citace: „U centralizovaného systému (ovládání elektrických spotřebičů) jsou vstupy (senzory) a výstupy (aktory) propojeny hvězdicově s centrálním řízením. Každý účastník (senzor, případně spotřebič) má vlastní spojení s centrálním řízením. Účastníci mohou vzájemně komunikovat jen prostřednictvím této centrály.“[8] Centrální řízení je v tomto případně zastoupeno centrální

jednotkou, katalogové označení CU2-01M[9]. Tato centrální jednotka je umístěna v rozvaděči a je možné její propojení s počítačem nebo vzdálené ovládání pomocí internetu či brány GSM.

Veškerá komunikace probíhá po sběrníkovém systému. Tato sběrnice je složena ze dvou vodičů. Jednotlivé komponenty systému můžeme libovolně připojovat kdekoliv na tuto sběrnici. Příklady struktur topologie sběrnice jsou liniová, hvězdicová, kruhová nebo stromová. Na obrázku 1, který je stažen přímo ze zdroje[5], je dobře viditelná sběrnice systému včetně znázornění různých komponentů.

2.3.1.3 Ovládání a konfigurace

Jak je již napsáno výše v textu, ovládat systém lze pomocí počítače nebo vzdáleně pomocí internetu nebo brány GSM.

Ke kompletní správě systému INELS, správu centrální jednotky i správu veškerých jednotek se využívá aplikace „Inels Designer & Manager“. V tomto softwaru se nastavují všechny akce systému a programuje se zde i komunikátor brány GSM.

The diagram illustrates the system bus architecture, divided into two main sections: **SENZORY** (Sensors) on the left and **AKTORY** (Actuators) on the right.

Central Components:

- CU2-01M**: Centrální jednotka (Central unit)
- GSM2-01**: GSM brána (GSM gateway)
- BPS2-02M**: Napáječ sběrnice přes tlumivku (Bus power supply through a choke)
- MI2-02M**: Externí master sběrnice (External master bus)

Inputs and Connections:

- Mobilní telefon (Mobile phone) connected via RS232 to GSM2-01.
- Zdroj 27 V (27V source) connected to BPS2-02M.
- LAN, PDA, PC, INTERNET, WWW, NTB connected to CU2-01M.

Interiérové prvky (Indoor elements):

- SENZORY:** SOPHY2 Multifunkční jednotka (Multifunctional unit), IART2-1, IDRT2-1 (Analogové a digitální pokojové termoregulatory), WSB-20, WSB-40, WSB-80 (Skupinové ovladače s krátkocestným ovládáním - Group controllers with short-range control), KEY2-01 Klávesnice EZS (EZS keypad).
- AKTORY:** SOPHY2 Multifunkční jednotka (Multifunctional unit), IR vysíláč (IR transmitter), Reprodukter (Reproducer).

Do instalační krabice (To installation box):

- SENZORY:** IM2-80B Jednotka vstupů (Input unit). Connected to: Senzor teploty (Temperature sensor), IR ovladač (IR controller), Ovládání hlasem (Voice control), Senzor úrovně okolního osvětlení (Ambient light level sensor), Pohybové čidlo (Motion detector).
- AKTORY:** LM2-11B Stmívací jednotka (Dimmer unit), SA2-01B Spínací jednobanňová jednotka (Single-channel switching unit), SA2-02B Spínací dvoubanňová jednotka (Two-channel switching unit), DAC2/04B Převodník digital analog (Digital-to-analog converter). Connected to: Žárovková svítidla (Incandescent lamps), Ovládací tlačítko/spínač (Control button/switch), DIML, Topení (Heating), Žářivková svítidla (Fluorescent lamps), Topení (Heating).

Moduly do rozvaděče (Modules to distribution board):

- SENZORY:** IM2-140M Jednotka vstupů (Input unit). Connected to: Senzor teploty (Temperature sensor), Ovládací tlačítko/spínač (Control button/switch), Ovládací tlačítko/spínač (Control button/switch), Detektor vody (Water detector), Pohybové čidlo (Motion detector), Hlásič požáru (Fire alarm), Zabezpečení objektu (Object security), Systémové poruchy (System faults).
- AKTORY:** SA2-02M Spínací jednotka (Switching unit), SA2-04M Spínací jednotka (Switching unit). Connected to: Žárovková svítidla (Incandescent lamps), Topení (Heating), Žárovková svítidla (Incandescent lamps), Rolety (Blinds), Ventilátor (Fan), Žárovková svítidla (Incandescent lamps).

Legend:

- Sběrnice CIB** (Solid blue line)
- Další sběrnice CIB** (Dashed blue line)

7

3 Aktuální stav dané problematiky

V této kapitole je nastíněna daná problematika v oblasti uživatelských rozhraní. Konfigurační systém budou využívat různí uživatelé, proto je potřeba se s problematikou tvorby uživatelských prostředí dobře seznámit. Jsou zde nastíněné nutné znalosti pro vytvoření tohoto konfiguračního systému.

3.1 Problematika uživatelského rozhraní

V této podkapitole budou popsány teorie a různé techniky, které se využívají k tvorbě správných uživatelských rozhraní. Pro tuto podkapitolu bylo využito zdrojů [10, 11].

3.1.1 Teorie

V dnešní době již prakticky nenalezneme prostředí, které by nebylo ovlivněno počítačem. Počítače se stali součástí každodenního života, jsou levné a usnadňují nám život. K tomu, aby nám tyto stroje usnadnili práci a práce byla co nejúčinnější, je nezbytné, aby uživatel mohl s tímto strojem komunikovat na co nejvyšší úrovni. Nesmíme ovšem opomenout, že s počítači nezacházejí jen uživatelé s profesionálními zkušenostmi. Je tedy nutné vytvořit komunikační most mezi uživatelem a počítačem, a aby tato komunikace probíhala co nejintuitivnějším způsobem.

Citace: „Intuitivní je pro člověka to, s čím se setkává běžně v průběhu celého života. Ačkoliv se během vývoje mnohdy uživatelská rozhraní orientovala spíše na stroj než na člověka a pro efektivní práci bylo od uživatele vyžadováno značné přizpůsobení se stroji, dnes již technika pokročila natolik, že trend směřuje k co nejpřívětivějšímu ovládání z hlediska uživatele. To znamená takové ovládání, které napodobuje úkony běžného života nebo simuluje komunikaci člověka s člověkem.“[11]

Počítače uživatelům mohou předávat informace přes výstupní periferie (monitor, tiskárna, plotter, reproduktor). Uživatelé mohou počítačům poskytnout informace přes vstupní periferie (myš, klávesnice, mikrofon, tablet, kamera, scanner). Z předávání informace od uživatele počítači se v dnešní době nejvíce využívá pohyb. Nejčastějšími zástupci jsou klávesnice a polohovací zařízení (myš, trackball, joystick, tablet).

Uživatelské rozhraní by mělo respektovat vnímavost lidského oka, stejně tak i lidské psychiky na barvy. Můžeme zde tedy s výhodou využít barev, které jsou pro lidské oko přirozené a působí uklidňujícím dojmem (zelená, modrá). Nebo můžeme využít barev agresivních (červená, černá, žlutá) pokud chceme uživatele na něco upozornit. Pokud bychom správně nezvolili barvy, mohlo by dojít u uživatele ke snížení pracovního výkonu a jeho nervozitě.

Citace: „Pro vytvoření kvalitního uživatelského rozhraní, které bude jak vyhovovat uživateli, tak i umožňovat rychlou a účinnou práci je třeba dodržovat jistá kritéria. Uživatelské rozhraní, které

má být úspěšné musí být ergonomické. To znamená, že velikost tlačítek pro ovládání musí být taková, aby nedocházelo k chybám zmíněným výše. Dále musí být skutečně vhodné pro ten typ práce, k jakému je určeno. Jiné ovládání má například textový editor a jiné grafický editor. Pokud vytváříme uživatelské rozhraní pro nějaký stroj, musíme brát v potaz i bezpečnostní hledisko, aby například uživatel nepřišel do styku s elektrickým proudem a podobně. Stejně tak musí být ovládací prvky na fyzickém zařízení takové, aby umožňovaly například snadnou údržbu. Rozhraní musí splňovat jistá ekonomická omezení. To znamená, že jeho vývoj a výroba nesmí přesáhnout takovou hodnotu, kdy už by výsledná cena byla pro zákazníka příliš vysoká. Je samozřejmé, že každý zákazník, zejména jedná-li se o neprofesionála vybírá hlavně podle prvního dojmu, který do značné míry utváří estetický dojem z rozhraní. Proto je nutné, aby náš produkt byl nejen kvalitní uvnitř, ale také aby se dobře prezentoval navenek.“[11]

Principy, které nám pomáhají v přirozených uživatelských rozhráních, se nazývají „Drag and Drop“ (posuň a pusť) a „Look and Feel“ (podívej se a pociť). Prvním principem byl „Drag and Drop“, až posléze se vyvinul princip „Look and Feel“. Je zde využíváno zvyků uživatelů na přímou manipulaci tak, jak jsou zvyklí z reálného světa. Toto řešení má dva kladné důsledky. Prvním důsledkem je, že s danou aplikací mohou pracovat i neškolení uživatelé. Toto je velmi významným faktorem k prolomení prvotních obav z aplikace. Druhým důsledkem je, že na uživatele neklademe takovou zátěž při práci. Uživatelé jsou zkušení z běžného života a mají již určité podvědomé návyky („vezme složku a hodí ji do koše“).

3.2 Jazyk HTML

Tato podkapitola nastíní problematiku jazyka HTML (HyperText Markup Language). Jedná se o jeden z jazyků pro vytváření stránek v systému World Wide Web, který nám umožňuje publikovat různé dokumenty na Internetu. Pro tuto podkapitolu bylo využito zdrojů [14, 15, 16].

3.2.1 Teorie

HTML (HyperText Markup Language), jak již z překladu můžeme vyčíst, je značkovací jazyk. Pomocí určitých značek můžeme definovat strukturu celého dokumentu – zde se hlavně bavíme o struktuře webových stránek. Jedná se tedy o jazyk pro specifikaci rozvržení různých dokumentů a hypertextových odkazů.

Značkovací jazyk znamená, že jeho zdrojový kód (text) obsahuje kromě textu s vlastním obsahem i instrukce pro zpracování. Tyto instrukce se zpravidla vyskytují v podobě značek (tzv. tagů). Tyto značky mají speciální význam a slouží k vymezení řídicích konstrukcí, pochopitelné je, že tyto řídicí konstrukce se v prohlížeči nezobrazí. Samotný HTML ovšem umí pouze zpracovat text, funkce jako např. výpočet součtu dvou čísel jsou pro něj zcela neznámé stejně jako různá grafická kouzla (grafy apod.).

Mezi hlavní výhody značkovacích jazyků můžeme pokládat to, že k jejich editaci nepotřebujeme žádné speciální programové vybavení, k úpravě nám postačí obyčejný textový editor. Pokud si budeme chtít tuto editaci ještě více usnadnit a zpříjemnit, můžeme použít jeden ze specializovaných programů, kde kromě samotné editace těchto dokumentů je můžeme i strojově generovat. Na druhou stranu ale používání značkovacího jazyka vyžaduje jisté znalosti, proto není možné si k těmto jazykům sednout a experimentovat bez toho, aniž bychom neznali aspoň základní syntaxi. V případě, že bychom chtěli vytvářet dokumenty i bez patřičných znalostí, využíváme k tomu nástroje WYSIWYG. WYSIWYG (zkratka z anglické věty „What you see is what you get“) je označení pro jistý způsob editace dokumentů, kdy zobrazená verze na obrazovce počítače odpovídá vzhledově totožné verzi výsledného dokumentu.

Ovšem editací dokumentů to pochopitelně nekončí a nebude Vám stačit pouze nějaký textový editor, příp. nějaký nástroj ze skupiny WYSIWYG. K tomu, abyste si vůbec mohli Vaši práci prohlédnout, budete potřebovat ještě webový prohlížeč, pokud tedy využijeme pouze služeb textového editoru. Bez webového prohlížeče nebude možné správně zobrazit to, co jsme pracně vytvořili v textovém editoru. Až webový prohlížeč udělá z vašeho výtvaru to, čeho chceme docílit, až v něm vidíme skutečný výsledek naší práce. Pokud bychom pracovali s nástroji skupiny WYSIWYG, tak ty již mají v sobě zakomponovaný prohlížeč a výsledek si můžeme přímo prohlédnout.

3.2.2 Standardizační organizace

Ve svých počátcích začínal jazyk HTML jako neformální specifikace, podobným způsobem začínalo i mnoho jiných oblíbených technologií, které v praxi využívalo jen nepatrné množství lidí. Postupem času se jazyk stával oblíbenějším a začalo s ním pracovat čím dál více lidí. Proto bylo nutné, aby se zavedli jisté formálnější prostředky, které by jasně definovali, sjednotili a usnadnili vytváření dokumentů. Jinými slovy bychom mohli říci, že bylo potřeba tento jazyk standardizovat.

3.2.2.1 Konsorcium W3C

Konsorcium W3C (World Wide Web Consortium) bylo vytvořeno za účelem standardizovat jednotlivé verze jazyka HTML. Členové této skupiny mají odpovědnost za patřičný standart, jeho návrh a též i jeho distribuci. Veškeré návrhy jsou volně přístupné na Internetu, aby jednotliví uživatelé mohli mít právo tyto návrhy volně komentovat. Tím dochází k zpětné vazbě od uživatelů a členové konsorcia W3C mohou případně, na základě připomínek uživatelů, tyto standarty případně upravit tak, aby vyhovovali co největšímu počtu uživatelů.

Toto konsorcium ovšem nemá na starosti pouze jazyk HTML, nýbrž odpovídá i za standardizaci všech ostatních technologií, které souvisí s World Wide Webem. Pracují tedy prakticky neustále, nýbrž do kategorie World Wide Web nespadá pouze jazyk HTML, ale též šablony kaskádových stylů CSS (Cascading Style Sheets). Dále jsem ještě patří standardy jazyka XML (Extensible Markup Language) a na dalších standardech souvisejících s adresováním dokumentů

na webu. A v poslední řadě taktéž podněcují i vznik případných dalších standardů, které mohou rozšířit stávající webové technologie.

3.2.2.2 Pracovní skupina IETF

Tato skupina IETF (Internet Engineering Task Force) má mnohem širší pole působnosti jak výše zmíněné konsorcium W3C. Odpovídá za definice veškerých aspektů internetových technologií. W3C tedy můžeme pokládat za jen jakousi malou oblast působení oproti skupině IETF. Nicméně tyto dvě skupiny, spolu ještě se skupinou ISO/IEC, úzce spolupracují.

Skupina IETF především definuje veškeré technologie Internetu. Tyto technologie definuje prostřednictvím oficiálních dokumentů. Tyto dokumenty jsou známe pod pojmem „žádost o komentář“ (Request For Comment), neboli RFC.

Tato skupina je otevřená, tudíž nevyžaduje žádné formální členství nebo členské požadavky. Všichni členové, i nejvyšší představitelé, jsou dobrovolníky a jejich odvedená práce je ve většině případů placena jejich zaměstnavateli případně sponzory.

Organizace IETF je organizována do nepřeborného množství skupin, kde každá skupina pracuje na určitém úkolu. V čele každé skupiny je předseda (předsedové) a pro každou skupinu jsou dána jistá ustanovení, ve kterých jsou přesně definované informace o tom, na čem má ta která skupina pracovat. Čím se má zabývat a co má vytvářet. Tyto skupiny můžeme dále rozdělit do několika oblastí podle zájmu, tzn. na jakém úkolu momentálně pracují. Takovými oblastmi jsou např. Internet, zabezpečení, TCP/IP aj. Každá oblast má pochopitelně i svého kontrolora, který dohlíží na chod celé skupiny.

3.2.3 Struktura dokumentu HTML

Základní struktura stránky může vypadat následovně:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">

<html>
  <head>
    Hlavička stránky
    <title> Titulek stránky </title>
  </head>

  <body>
    Obsah stránky
  </body>
</html>
```

- **!doctype** – obsahuje informaci o verzi použitého HTML a typu DTD (Document Type Definition). DTD. Říká prohlížeči, jaké jsou obsaženy elementy v dokumentu a jak s nimi má pracovat, nicméně tento element není součástí HTML

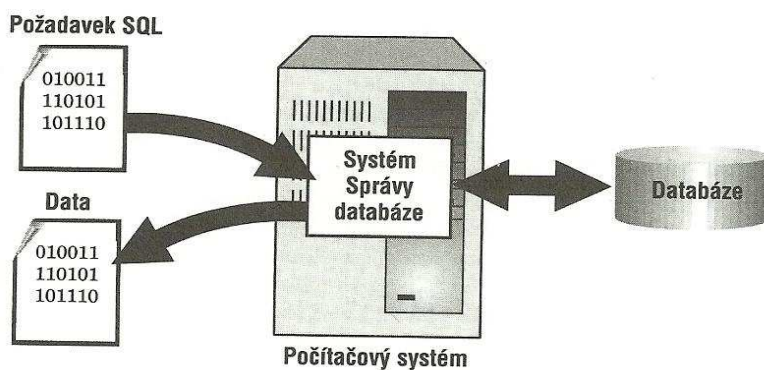
- **html** – začíná a končí celý dokument a veškerý další obsah musí být mezi těmito tagy
- **head** – jedná se o hlavičku dokumentu, která se sice nezobrazuje, ale obsahuje důležité informace o stránce. V hlavičce se mohou vyskytovat další značky: `title`, `link`, `meta`, `script`
- **title** – obsahuje titulek stránky, který se zobrazuje v prohlížeči v horní liště, ale má také jiné funkce jako např. zobrazení nadpisu ve vyhledávačích nebo indexace stránky pro vyhledávací roboty
- **link** – využívá se k provázání stránky s externím souborem, většinou se jedná o soubor s kaskádovým stylem (CSS).
- **meta** – obsahuje informaci o dokumentu, použitých klíčových slovech nebo kódování
- **body** – tělo dokumentu, mezi těmito tagy se nachází veškerý zobrazovaný obsah stránky
- **<!-- poznámka -->** - veškeré informace zapsané místo slova „poznámka“ nejsou zobrazeny ve výsledné webové stránce. Do těchto značek se nejčastěji píše komentáře k jednotlivým částem kódu webové aplikace a tato poznámka může být umístěna v jakékoli části dokumentu.

3.3 Jazyk SQL

V této podkapitole si nastíníme jazyk SQL jako hlavní dotazovací jazyk. Tento jazyk je využíván pro práci s daty v relačních databázích. Pro tuto podkapitolu bylo využito zdrojů [17, 18, 19, 20].

3.3.1 Teorie

Citace: „Jazyk SQL je nástroj pro organizování, správu a získávání dat uložených v počítačové databázi. Zkratka SQL znamená strukturovaný dotazovací jazyk (Structured Query Language). Jak vyplývá z názvu, SQL je počítačový jazyk, který se používá pro komunikaci s databází. Ve skutečnosti SQL pracuje s jedním specifickým typem databáze, který nazýváme relační databáze.“[17]



Obrázek 2: Ukázka způsobu práce SQL [17]

Citace: „Když potřebujeme z databáze získat data, napíšeme požadavek v jazyce SQL. Databázový systém jej zpracuje, načte data a vrátí odpovídající informace. Proces, kdy požadujete data z databáze a systém vám vrátí výsledek, se nazývá databázový dotaz (query), a odtud plyne samotný název – strukturovaný dotazovací jazyk.

Název strukturovaný dotazovací jazyk je ve skutečnosti nevhodné pojmenování. Tak především, jazyk SQL představuje mnohem více než jen pouhý dotazovací nástroj, i když to byl jeho původní účel a získávání dat patří stále mezi jeho nejdůležitější funkce. Lze jej taktéž použít k řízení všech funkcí, které databázový systém svým uživatelům poskytuje, včetně:

- **Definice dat.** SQL dovoluje uživateli definovat strukturu a organizaci uložených dat spolu s definicí jejich vzájemných vztahů.
- **Získávání dat.** SQL umožňuje uživateli nebo aplikačnímu programu z databáze uložená data získávat a používat.
- **Manipulace s daty.** SQL umožňuje uživateli nebo aplikačnímu programu databázi aktualizovat přidáváním nových dat, odstraňováním starých nebo změnou již dříve uložených dat.
- **Řízení přístupu.** SQL lze použít k omezení schopnosti uživatele data číst, přidávat a modifikovat, čímž je lze chránit před neautorizovaným přístupem.
- **Sdílení dat.** SQL se používá ke sladění sdílení dat mezi více uživateli a k zajištění toho, že se uživatelé mezi sebou navzájem neruší.
- **Integrita dat.** SQL definuje v databázi omezení integrity, čímž ji chrání před porušením, které může vzniknout kvůli neúplným aktualizacím nebo systémovým selháním

SQL je proto komplexní jazyk pro řízení a ovlivňování databázového systému.“[17]

3.3.2 Historie

V 70.let 20.století se tento jazyk nazýval *SEQUEL* (Structured English Query Language) a tento název si ponechal až do konce 70.let. Vznikl v laboratořích firmy IBM a byl použit v jejich firemním systému, který nesl název *R*. A prvotním cílem bylo vytvořit takový jazyk, aby se při syntaktické tvorbě příkazů tyto příkazy co nejlépe podobaly přirozenému anglickému jazyku.

K vývoji tohoto jazyka se postupně přidala celá řada dalších firem, mezi nejznámější bezpochyby byla firma Relational Software, Inc. (dnešní Oracle Corporation). Ta v roce 1979 uvedla na trh svoji platformu relační databáze *Oracle Database*. O dva roky později přinesla na trh firma IBM nový systém s označením SQL/DS, nicméně nový systém na sebe nenechal dlouho čekat. Již v roce 1983 vydala stejná systém DB2. Ve všech těchto systémech byla použita varianta jazyka SEQUEL, tento název byl poté zkrácen do dnešní známé podoby SQL.

V roce 1986 byl přijat vůbec první standart, který nesl označení *SQL86*. Tento standart byl vůbec první a tudíž asi nikoho nepřekvapí, že obsahoval jisté nedostatky, které samozřejmě musely být opraveny. Proto byla v roce 1992 vydána nová verze standardu s označením *SQL92*, který je v oblasti relačních databází standardem dodnes.

3.3.3 Příkazy jazyka SQL

V hlavní části jazyka SQL můžeme nalézt přibližně 40 příkazů. Každý příkaz v SQL začíná klíčovým slovem (nebo také slovesem), které popisují co daný příkaz provádí. Za klíčovým slovem se vyskytuje klauzule, která specifikuje data, se kterými má příkaz pracovat. Pro představu zde některé z těchto nejznámějších příkazů uvedu i s popisem:

- Manipulace s daty:
 - SELECT vyhledá data v databázi
 - INSERT vloží do databáze nová data
 - DELETE odstraní data z databáze
 - UPDATE změni stávající data databáze
- Definice dat:
 - CREATE vytvoří nový objekt v databázi (např. tabulku)
 - DROP odstraní objekt z databáze (např. tabulku)
 - ALTER změni strukturu objektu v databázi (např. tabulky)
- Řízení přístupu:
 - GRANT přidělí přístupová práva uživateli
 - REVOKE odebere přístupová práva uživateli
- Řízení transakcí:
 - COMMIT ukončení aktuální transakce
 - ROLLBACK zrušení aktuální transakce
 - SET TRANSACTION nastavení vlastností transakce (její druh a stupeň izolace)
- Programové SQL:
 - DECLARE definice kurzorové tabulky pro dotaz
 - EXPLAIN popis plánu přístupu k datům pro dotaz
 - OPEN otevření kurzorové tabulky pro načtení výsledků z dotazu
 - FETCH načtení řádku z výsledku dotazu
 - CLOSE zavření kurzorové tabulky
 - PREPARE příprava příkazu na dynamické provedení
 - EXECUTE dynamické provedení příkazu
 - DESCRIBE popis připraveného dotazu

3.4 Jazyk PHP

V této podkapitole bude popsán jazyk PHP jako hlavní programovací jazyk pro vývoj této aplikace. Bylo zde využito poznatků ze zdrojů [12, 13, 23].

3.4.1 Teoretický rozbor

PHP („*PHP: Hypertext Preprocessor*“, původně *Personál Home Page*) je programovacím jazykem. Jedná se o skriptovací serverový jazyk, který umožňuje vytváření webových dynamických stránek. Jeho nejčastější začlenění je ve strukturách jazyků HTML (*HyperText Markup Language*) a XHTML (*Extensible HyperText Markup Language*) a propůjčuje jim tak schopnosti generovat požadovaný obsah. Což dělá z obyčejných statických stránek dynamické dílo.

Syntaxe jazyka je podobná programovacím jazykům Perl, C, Pascal a Java. Jazyk PHP, stejně jako třeba Perl, je nezávislý na platformě, vytvořené skripty fungují jak na operačních systémech Windows, tak i Linux a ostatních. Podpora mnoha knihoven dělá z jazyka PHP mocný nástroj – knihovny pro práci se soubory nebo pro přístupy k databázovým systémům (např. MySQL, Oracle, PostgreSQL). Pochopitelně zde lze najít i podpory pro zpracování textu nebo grafiky.

Podpora mnoha internetových protokolů (HTTP, SMTP, FTP, IMAP, POP3...) a díky své jednoduchosti se stal jazyk PHP velmi oblíbeným nástrojem. Svoji zásluhu v této oblibě také hraje vlastnost kombinace různých programovacích jazyků.

Skripty PHP jsou prováděny na straně serveru a k uživateli je přenesen až samotný zpracovaný výsledek. Typický skript PHP obsahuje jednak části HTML kódu, tak i části naprogramovaného kódu. Když tedy bude prohlížeč požadovat nějakou webovou stránku, tak server tuto stránku nejprve sestaví a pak až pošle zpět uživateli do webového prohlížeče. Jednotlivé kroky by se dali ve stručnosti popsat třemi kroky, tak jak je server zpracovává:

- kusy HTML kódu zanechá beze změny
- části naprogramované jazykem PHP provede
- výsledek zkomponuje a předá prohlížeči

Tato koncepce fungování je velmi mocná. Server může provést jednu nebo i několik tisíc operací a výsledek poslat jako obyčejnou HTML stránku prohlížeči.

Dnes můžeme slyšet pojem PHP ve spojení s jiným pojmem a to s LAMP. LAMP je zkratka, která označuje sadu „open source“ softwaru pro vytváření dynamických webových stránek. Zahrnuje operační systém Linux, webový server Apache, databázový systém MySQL a programovací jazyk PHP.

3.4.2 Historie

V roce 1994 byl napsán jednoduchý systém pro počítání přístupu ke stránkám. Tento skript napsal Rasmus Lerdorf a jednalo se o skript v jazyce PERL. Po nějaké době byl skript přepsán do jazyka C, vzhledem k tomu, že perlowský kód dosti zatěžoval server. Později ve stejném roce byla vydána sada těchto skriptů pod názvem „Personal Home Page Tools“ (zkráceně PHP).

V polovině roku 1995 se nástroj „Form Interpreter“ (od stejného autora) právě spojil se zmiňovaným PHP a vznikl tak systém PHP/FI. V roce 1997 vznikla verze PHP/FI 2.0, tento systém se postupně rozšířil do celého světa a získal si celosvětovou proslulost.

V roce 1997 byl ještě tento systém kompletně přepsán. O přepsání se postarali Andi Gutmas a Zeev Suraski. Pokoušeli se použít PHP/FI pro vývoj komerčně založených aplikací, ale shledali jej jako poddimenzovaný. Došlo tedy k jeho kompletnímu přepsání a výsledkem bylo PHP verze 3. Poté došlo k opětovnému přepisování jádra PHP, aby se zvýšil výkon pro složité aplikace, a v roce 1999 vydali Zend Engine.

V roce 2000 již toto jádro bylo zakomponováno a objevila se verze PHP4. O čtyři roky později již byla představena verze PHP5, která již běží na novém jádru Zend Engine II. PHP5 již obsahuje nové rysy jako vylepšení objektově orientovaných programů nebo množství výkonných vylepšení a je nejpoužívanější verzí v dnešní době.

3.4.3 Vkládání PHP do HTML

Soubory napsané v jazyce PHP obvykle mají příponu *php*, případně *php3*, *php4* nebo *phtml*. Vzhledem k tomu, že PHP slouží pro vytváření webových stránek, nikoho jistě nepřekvapí, že se tyto skripty dají volně vkládat do kódu HTML. Jen je potřeba je oddělit od ostatního kódu HTML.

K oddělení kódu PHP od kódu HTML nám slouží značky „<?php“, která ukazuje na začátek zápisu kódu php a „?>“, která tento kód uzavírá. Všechn kód, který je napsaný mezi těmito dvěma značkami je brán jako kód PHP, ostatní kód je považován za HTML.

Existují různé způsoby zápisu značek php, pro krátký styl zápisu postačuje zapsat pouze značky „<?“ a „?>“. Nicméně tento způsob zápisu zřejmě na většině serverů funguje, nicméně dostupnost již není nikterak zaručena, takže klidně může nastat situace, že skript nebude fungovat. K tomu aby tento krátký zápis adekvátně fungoval, bude zapotřebí povolit *short_tags* v konfiguračním souboru. Asi nejlepším způsobem zápisu je styl, kdy zapíšeme značky „<?php“ a „?>“. Tento styl má garantovanou dostupnost na všech serverech, protože jej administrátor nemůže vypnout.

3.4.4 Popis struktury PHP

3.4.4.1 Příkazy

Veškeré příkazy se musí nacházet mezi značkami `<? a ?>` a každý příkaz musí být oddělen středníkem.

3.4.4.2 Komentáře

Komentáře jsou části kódu, které program při zpracování ignoruje. Slouží pouze k lepší orientaci v kódu, programátor si tak může okomentovat jednotlivé části a lépe se tak vyznat v kódu. Existují zde jednořádkové nebo víceřádkové komentáře. Jednořádkový komentář začíná dvojicí lomítek `„/“` nebo křížkem `„#“`, oproti tomu víceřádkové komentáře začínají kombinací lomítka a hvězdičky `„/*“` a je ukončen obrácenou kombinací, tedy hvězdička a lomítko `„*/“`.

3.4.4.3 Proměnné

Stejně jako v každém jiném jazyce, i zde mohou proměnné nabývat různých typů. Ale vždy musí začínat znakem dolar `„$“` následující posloupností písmen, číslic a podtržítek. Je ovšem zapotřebí si dávat pozor na case-sensitive, takže proměnná `„$A“` a `„$a“` jsou dvě různé proměnné.

3.5 JavaScript

V této kapitole si stručně vysvětlíme jazyk JavaScript, který pro tento systém bude taktéž využit. Využijeme jej jako vedlejší jazyk, který nám bude hlídat jisté změny ve webovém systému. Využili jsme zde zdrojů [24, 25, 26, 28].

3.5.1 Teorie

JavaScript je objektově orientovaný programovací jazyk. Jedná se o multiplatformní jazyk. To znamená, že jazyk JavaScript je funkční pod různými operačními systémy, stejně jako například jazyk PHP. Ale není to pouze multiplatformnost, kterou má JavaScript s PHP společné, taktéž je tento jazyk interpretovaný, což znamená, že se kód v tomto jazyce nemusí překládat. Největší využití tento jazyk nachází v oblasti Internetu. Zapisuje se přímo do kódu jazyka HTML. Jedná se o bezesporu výhodu, nepotřebujeme totiž k práci žádné složité vývojářské nástroje, nýbrž nám k vytváření skriptů postačí pouhopouhý textový editor.

Při používání JavaScriptu nesmíme zapomenout, že se jedná o klientský skript. To znamená, že veškeré informace zpracované JavaScriptem se zpracovávají na straně klienta. To tedy znamená, že veškeré operace se posílají do prohlížeče a teprve tam se vykonají. Narozdíl např. od jazyka PHP, který spadá do kategorie serverových jazyků. Neměli bychom ale ani zapomenout na webový

prohlížeč který používáme, ne všechny webové prohlížeče korektně zobrazují skripty napsané v JavaScriptě.

Asi nejjednodušším příkladem použití JavaScriptu jsou hodiny umístěné na webových stránkách. JavaScript můžeme použít jako „vyhodnocovač“ dat webových formulářů, případně s ním můžeme tvořit všemožné prvky, které nám ožíví webové stránky.

Velmi důležitou záležitostí je, že JavaScript je tzv. CaseSensitif. Proto musíme velmi dobře dodržovat jistá pravidla v psaní kódu. Jen pro představu, `document.write` není to stejné jako `document.WRITE`, toto pravidlo je nutné dodržovat, skript by jinak nefungoval. Jinak veškeré programy, které jsou napsané v jazyce JavaScript se nazývají skript (skripty). Ty mohou být obraženy buď přímo v kódu stránky a nebo mohou být externě načítány do stránky z souboru. Soubory se skripty jazyka JavaScript se poznají podle přípon těchto souborů, přípony jsou `.js` nebo `.jse`.

Protože je JavaScript klientským skriptovacím jazykem, jsou zde jisté omezení. JavaScript totiž neumí přistupovat k žádným souborům, teda kromě cookies, a ani k žádným systémovým objektům. Taktéž můžeme říct, že i ukládání dat je pro něj taktéž jednou neznámou kapitolou. Je to celkem logické, vzhledem k bezpečnosti jsou tyto opatření nezbytná. Ovšem je pravda, že pokud si uživatel nebude přát využívat služeb JavaScriptu, může ho zakázat.

3.6 Kaskádové styly CSS

Tato kapitola popisuje kaskádové styly. Využili jsme zde informací ze zdrojů [14, 21, 22, 27].

3.6.1 Popis

Kaskádové styly (*Cascading Style Sheets* - zk. **CSS**) jsou jazyky, které popisují způsob zobrazení stránek napsaných ve značkovacích jazycích (HTML, XML aj.). Vzhled je definován odděleně od kódu značkovacího jazyka, což můžeme pokládat za velkou výhodu. Takto oddělený vzhled poté můžeme snadno modifikovat. Vzhled může být taktéž definován centrálně pro velké množství stránek. K tomuto účelu se nejvíce hodí a používají externí soubory s definicí vzhledu CSS. Bylo by velmi nepraktické na každé stránce HTML zvlášť definovat vzhled a neustále jej překopírovávat do jednotlivých stránek. Toto řešení je sice možné, ale při tvorbě webu s větším počtem stránek dosti nepraktické. K tomuto účelu se právě využívají externí soubory, které obsahují definici vzhledu.

CCS bylo navrženo organizací W3C (popis viz Standardizační organizace, kap. 3.2.2.) a prvotním autorem byl muž se jménem Håkon Wium Lie. Doposud byly vydány jen dvě úrovně tohoto jazyka, CCS1 a CCS2. CCS1 spatřilo světlo světa v roce 1996, ovšem jako asi každá prvotní verze i tato měla jisté problémy s kompatibilitou. Webové prohlížeče Netscape4 a IE4 měli pouze neúplnou podporu, což se ovšem zlepšilo v následujících verzích, v Netscape5, Mozilla a IE5.x. O dva roky později, v roce 1998, vyšla nová verze CCS s označením CCS2, měla slušnou podporu v Mozilla. Podmnožinu CCS2 se v dnešní době využívá prakticky všude. V roce 2009 vyšla revize verze CCS2

s označením CCS2.1, byly zde opraveny některé chyby, bylo přidáno pár nových vlastností a byly odstraněny některé nepodporované vlastnosti. Nicméně již se pracuje na návrhu verze CCS3.

3.6.2 Syntaxe jazyka

Syntaxe kaskádových stylů se skládá z posloupnosti pravidel. Každé pravidlo obsahuje dvě záležitosti, selektor a deklaraci vlastností:

```
selektor { deklarace vlastností }
```

Ukázkový příklad:

```
tělo {  
    background-color: blue;  
    color: red;  
}
```

Celý blok jsem pojmenoval „tělo“, jedná se o pravidlo selektoru a v závorkách je samotná deklarace. Řádek „background-color: blue“ nastavuje barvu pozadí stránky na modrou. V tomto případě je „background-color“ vlastnost a „blue“ je hodnota této vlastnosti.

Selektor zde určuje, na jaké části se bude dané pravidlo vztahovat. A vlastnost selektoru definuje vizuální vlastnost dotyčného elementu.

3.6.2.1 Selektory

Citace: „CSS definuje mnoho různých selektorů, které obvykle můžeme kombinovat. Mezi základní patří:

- `body` – Tyto deklarace budou platit pro všechny výskyty elementu `body`.
- `body p` – Tyto deklarace budou platit pro všechny elementy `p`, které se nachází v elementu `body`, v jakékoliv hloubce.
- `body>div` – Tyto deklarace budou platit pro všechny elementy `div`, které jsou dětmi elementu `body`. To znamená, že pokud bychom měli element `div`, který se nachází v `<body><blockquote><div>...`, tyto deklarace by pro něj neplatily, protože tento `div` není přímým potomkem elementu `body`.
- `.trida` – Tyto deklarace budou platit pro všechny elementy, které mají v HTML nastavenou třídu `trida`. To se provádí pomocí HTML atributu `class`.
- `#id` – Tyto deklarace budou platit pro všechny elementy, které mají v HTML nastavený identifikátor `id`. To se provádí pomocí HTML atributu `id`.
- `sel1, sel2, sel3` – Selektory můžeme seskupovat pomocí čárek. Následující deklarace pak budou platit pro všechny selektory.“ [21]

3.6.2.2 Připojení CSS stylů do stránky HTML

Připojení CCS stylů do stránky HTML můžeme provést pomocí několika technik. První z nich je přímé vložení stylu pomocí atributu `style`. Dalšími způsoby je vložení šablon stylů na úrovni dokumentu a propojení s externími šablonami stylů.

První možnost, tedy přímé vložení stylu, představuje tu nejjednodušší variantu spojení. Stačí pouze do značky (tagu) zapsat atribut *style* a přiřadit mu požadovanou vlastnost (Příklad: `<h2 style="color: red"> AHOJ </h2>` vypíše text „AHOJ“ v červené barvě). Nicméně tento příklad vkládání není příliš vhodný, vložené styly jsou v podstatě volně roztroušené, a proto se můžou špatně udržovat.

Druhý způsob (šablony na úrovni dokumentů) představuje zápis pravidel do záhlaví dokumentu, mezi značky `<head>` se zapíše značka `<style>` a jeho příslušná ukončovací značka `</style>`. Tyto značky právě uvozují definici stylů (Příklad: `<style type="text/css"> styl1 {color: red} </style>` a pokud poté vložíme do těla dokumentu `<styl1> AHOJ </styl1>` vypíše se text „AHOJ“ v červené barvě).

Třetím způsobem vkládání je propojení s externí šablonou. Největší výhodou je, že tuto šablonu můžeme mít uloženou kdekoliv na síti a pak jen jednoduše načíst do dokumentu HTML. Propojení se děje pomocí značky `<link>` (Příklad: `<link rel=stylesheet type="text/css" href="./style.css">` načte soubor *style.css*, který je uložený ve stejné složce jako soubor HTML a obsahuje příslušné selektory s vlastnostmi). Pokud zaměníme *./style.css* za *www.xxx.cz/styl/style.css*, nebo jinou adresu, můžeme načíst soubor CSS z jiného uložení ze sítě.

4 Návrh aplikace

Při návrhu aplikace jsem se řídil požadavky, které si na aplikaci klade firma ELKO EP s.r.o. Software budou využívat koncový uživatelé a tudíž hlavním cílem bude jednoduchá přístupnost tohoto konfiguračního prostředí.

Vzhledem k tomu, že se bude jednat o webový konfigurační nástroj, bude hlavní podmínkou využití, na straně uživatele, nutnost mít nainstalovaný webový prohlížeč a přístup k síti Internet. I když nutností webového prohlížeče se v dnešní době nemusíme příliš zabývat, zpravidla každý operační systém již v základním vybavení nějaký webový prohlížeč obsahuje. Například v systémech Microsoft Windows najdeme aplikaci Internet Explorer. V systémech Linux, distribuce Ubuntu, je pro prohlížení webových stránek využito služeb Mozilla Firefox.

Dalším důležitým bodem bude nutnost podpory PHP skriptů na serveru. Tato aplikace bude těchto skriptů využívat a bez této podpory by nebylo možné správné zobrazování výsledků jednotlivých úkonů.

4.1 Obsah aplikace

Aplikace bude obsahovat seznam všech jednotek systému INELS. K tomuto úkonu bude potřeba prostudovat jak technický katalog INELS[9], tak i ceník jednotlivých komponent[12]. Aplikace dále bude obsahovat seznam všech zařízení, které v systému mohou být (jedná se o databázi zařízení, světel, zásuvek, spotřebičů atd.). Pro jednotlivé typy zařízení bude muset být jasně definované o jaké funkce se bude jednat, např. stmívač, spínač, časovač atd.

Mezi dalšími prvky aplikace najdeme konfiguraci objektu. Pod tímto pojmem si můžeme představit typ budovy např. obytná budova, výrobní budova atd. Po výběru typu budovy bude uživateli nabídnut výběr jednotlivých podlaží v daném objektu, např. obytný dům bude obsahovat podlaží s názvem „sklep“. Dále bude uživateli nabídnuto, aby si zvolil jaké pokoje v jednotlivých podlažích má. U každého pokoje si bude moci přímo napsat počet zařízení, které bude chtít např. spínat. Mezi poslední složky se řadí volba provedení, zda-li preferuje zapojení do rozvaděče nebo do BOX krabice. Dále zde bude volba grafického provedení krytů a rámečků a v poslední řadě zde bude výpis. Výpis bude obsahovat typ jednotky, počet těchto jednotek a patřičnou cenu.

4.2 Vzhled aplikace

Co se týče samotného vzhledu aplikace, zde nebude příliš možná vlastní vymyšlená grafická stránka. Veškerá grafická stránka se bude řídit a měla by být plně v korekci se vzhledovým manuálem firmy, tzv. „Corporate Design Manual“.

„Corporate Design Manual“ je systematický soubor, kde jsou jasně definovaná pravidla pro používání loga, firemních barev, písma, ochranných známek, propagačních materiálů atd. V tomto manuálu nalezneme tedy veškeré informace, které utváří firemní grafickou identitu. Firmy zavádějí tento manuál, aby tím srozumitelně nastavili pravidla požadované na svůj vzhled a aby tím ulehčili práci těm, kteří s nimi pracují. Tento manuál také může sloužit jako odlišení se od firem s podobným zaměřením.

4.3 Funkce aplikace

Uživatel si bude vybírat zařízení ze seznamu zařízení (databáze zařízení, světla, vypínače...) a bude jim zároveň přiřazovat i jejich funkci v systému (stmívač, spínač, regulátor...). K jednotlivým zařízením ze seznamu zařízení se budou přiřazovat patřičné jednotky INELS ze seznamu všech jednotek INELS.

Aplikace bude hlídat celkový počet jednotek na sběrnici. Vytížení sběrnice totiž není neomezené a bude potřeba tuto skutečnost ošetřit. Pokud dojde k překročení některého z parametrů, bude přidána patřičná jednotka, např. při maximálním vytížení obou sběrnic řídicí jednotky bude přidán modul, který maximální počet jednotek navýší. Dalším hlídacím bodem bude spotřeba a dimenze napájecích zdrojů, při překročení maximálního zatížení zdroje bude přidán další zdroj.

Aplikace by měla mít i určité administrátorské prostředí, aby bylo možné přidávat jednotky do databáze, případně některé jednotky z databáze odebírat nebo upravovat.

Výstupem aplikace bude výpis vybraných jednotek a zařízení s příslušnou cenou.

5 Implementace aplikace

Úkolem mé bakalářské práce bylo prostudovat literaturu týkající se webových konfiguračních nástrojů a systému INELS společnosti ELKO EP s.r.o. Po nastudování všech nezbytných informací bylo dalším úkolem navrhnout vhodný konfigurační nástroj, který by splňoval požadavky, které na tento systém byly kladeny. Dále bylo nutné nastudovat možnosti jednotlivých implementačních jazyků vhodných pro tvorbu webových nástrojů. V této kapitole bude popsán výběr jednotlivých jazyků a funkce, které v systému mají. A bude zde také popsána samotná aplikace a její požadavky.

5.1 Požadavky aplikace

K vývoji této aplikace bylo využito webového serveru v balíčku EasyPHP. EasyPHP je balík obsahující služby PHP, PHPMyAdmin, MySQL a Apache. Jeho nespornou výhodou je jednoduchá instalace a nastavení pod operačním systémem Microsoft Windows. Já jsem tento program využil ve verzi EasyPHP 5.3.0, která byla nejnovější verzí v době zahájení práce na této aplikaci, nicméně dnes je verze tohoto programu již novější. Program je volně dostupný na stránkách www.easyphp.org. Program EasyPHP 5.3.0 obsahuje služby Apache verze 2.2.13, dále je zde služba MySQL ve verzi 5.1.37 a také PHP ve verzi 5.3.0.

Důvodem této volby bylo, že webový server jsem měl instalovaný přímo na osobním počítači a mohl jsem si jej tudíž lehce přizpůsobit. A nebylo tudíž ani třeba být neustále připojený k nějakému vzdálenému webovému serveru a kontrolovat funkčnost php skriptů. Nicméně i přes vlastní webový server jsem si zřídil účet i na serveru „Internet Centrum“ (www.ic.cz), kde jsem též testoval funkčnost všech implementovaných funkcí. Jedná se o server, který poskytuje web hosting zdarma.

K bezproblémovému chodu aplikace nejsou potřeba žádné speciální aplikace. Aplikace bezproblémově běží na každém webovém serveru s PHP ve verzi 5 a databází MySQL. Jediný problém přenositelnosti je, že pokud umístíme aplikaci na nějaký server, je nutné změnit přihlašovací údaje v souboru config.php a vytvořit databázi pomocí souboru inels.sql. Další nutností je mít povolenou funkci JavaScriptu na svém webovém prohlížeči. Aplikace obsahuje funkce JavaScriptu a bez povolení této funkce by tyto funkce nefungovali tak jak by měli. Aplikace by se tedy stala nepoužitelnou. Ostatní potřeby jsou již běžně dostupné ve webových prohlížečích, zpracování jazyka HTML a podpora kaskádových stylů.

5.2 Vývoj aplikace

Aplikace byla vyvinuta na jazyce PHP, který slouží jako stěžejní jazyk pro tento konfigurační nástroj. Tento jazyk zde zpracovává veškerá data a předává jejich výsledky. Data jsou zpracovávána jak z pohledu různých funkcí, např. sčítání čísel, tak i z pohledu dotazování a zpracovávání těchto dotazů s databází MySQL. Aplikace prošla jistým vývojem, při kterém byla zvolena, z mého pohledu, nejvhodnější varianta konečné implementace. Pro vývoj bylo využito taktéž poznatků a možností, které jsem načerpal prostudováním produktu Theben Luxor. Tyto poznatky byly velmi cenné a vývoj mé aplikace tím dostal konečnou podobu.

Velký pokrok též zaznamenala databáze MySQL, která oproti původní verzi dosti zeštíhlela. V původním návrhu této databáze bylo využíváno nepřeberného množství tabulek a údajů, které se v určitém čase již staly velmi nepřehledné a muselo dojít k jejich optimalizování. Optimalizace byla vedena pro přímou potřebu aplikace a vynechání zbytečných údajů, které jak se stejně později ukázalo, nejsou pro chod aplikace důležité. Nicméně pro případný další vývoj aplikace by jistě své uplatnění tyto data našly.

5.2.1 Vývoj uživatelského rozhraní aplikace

Vývoj grafické části uživatelského prostředí prošel zřejmě nejdelší etapou. V prvotní verzi bylo jen velice jednoduché grafické prostředí, které obsahovalo jen několik málo grafických prvků. Taktéž barevnost velmi strádala, když převládající barvou byla jen jedna základní barva, která v jistých částech byla změněna za jinou barvu. A to prakticky jen z důvodu lepšího zviditelnění určité části stránky. Toto prostředí bylo tedy velice jednoduché, ale bohužel postrádalo tu patřičnou uživatelskou „líbivost“ a přívětivost.

Došlo tedy k úpravě toho vzhledu a další fázi již byly webové stránky o něco přívětivější. Skládaly se ze záhlaví, které obsahovalo logo systému INELS, další část obsahovala menu s jednotlivými položkami a části, která obsahovala hlavní část celého konfiguratoru. Strukturovaný nástin můžete vidět níže na obrázku 3.

Tato verze ovšem neměla dlouhého trvání. Do webových stránek jsem zapojil kaskádové styly CSS a vzhled stránek se ihned stal ještě více přívětivějším a líbivějším. A tato verze se již také stala konečnou grafickou podobou tohoto systému. Ukázku můžete vidět na obrázku 4.

Na tomto obrázku je již i patrné rozdělení celé aplikace. V hlavním lze vidět položku „Domovská stránka“, která obsahuje jistý úvod pro tento systém. Pod druhou položkou „Inels.cz“ je odkaz na oficiální stránky INELS. Pod záložkou „Konfigurator“ se ukrývá samotná hlavní aplikace a tedy hlavní předmět této bakalářské práce. Čtvrtá položka s nápisem „Nápověda“ obsahuje nápovědu jak, samo o sobě, název napovídá. Pod poslední položkou „Přihlášení“ je ukryto administrátorské rozhraní pro úpravu databáze MySQL.



Obrázek 3: Druhá fáze vývoje grafického rozhraní



Obrázek 4: Výsledná podoba webové aplikace

5.3 Rozdělení aplikace

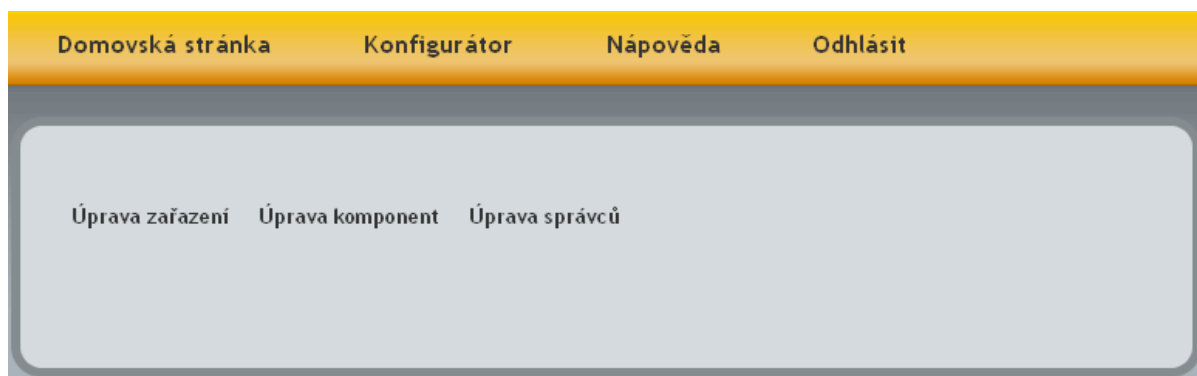
Aplikaci jsem rozdělil do tří hlavních částí. První část můžu nazvat jako uživatelské informace a je složena ze dvou stránek. Jedná se o dvě položky v menu. První položkou je „*Domovská stránka*“, která obsahuje úvod a jistý obecný popis. V druhé položce s názvem „*Nápověda*“ nalezne uživatel veškeré nutné informace pro pochopení systému a možnostmi volby. Jsou zde popisy jednotlivých kroků a úkonů, které zde může vykonat.

Druhá část aplikace obsahuje administrátorské rozhraní, které se ukrývá pod názvem položky „*Přihlášení*“. Pro přístup do toho systému je nutné znát uživatelské jméno a heslo. Obsahuje části pro úpravu položek databáze MySQL. Pro administrátory s vyššími právy je zde i možnost úpravy ostatních podřízených členů.

Třetí část obsahuje samotný základní konfigurator systému INELS. Je rozdělen do několika kroků z nichž každý krok má svůj jedinečný význam pro další vyhodnocení.

5.3.1 Administrátorské rozhraní

Jedná se o typické administrátorské rozhraní, které můžeme nalézt ve spoustě jiných aplikací, kde se určitým způsobem různě upravují data v databázi. Po autorizovaném přihlášení se Vám objeví podnabídka s možnostmi úpravy jednotlivých údajů (obrázek 5). Takto kompletní nabídku ovšem vidí pouze administrátoři s vyššími právy. Ostatní administrátoři mohou vidět pouze dvě položky a to „*Úprava zařazení*“ a „*Úprava komponent*“. Pod každou z možností se nacházejí tři hlavní funkce. První z funkcí je odstranění dané komponenty, další možností je úprava jednotky a v neposlední řadě také její vytvoření. Při otevření libovolné možnosti je zobrazen seznam dostupných údajů. Typ údaje je rozkliknutelná možnost pro vypsání detailů tohoto údaje.



Obrázek 5: Administrátorské volby

Každý detailní výpis je proveden jako formulář pro úpravu. To znamená, že administrátor nemusí přepínat při úpravě na jinou stránku, ale změnu provede přímo, což pokládám za velkou výhodu. A tuto změnu potvrdí tlačítkem „*Upravit komponentu*“ v případě volby, že chce změnit

hodnoty nebo název určité komponenty. V případě pouhého prohlížení se touto volbou nemusí zabývat. Na obrázku 6 je ukázka jednoho takového výpisu.

DETAILY KOMPONENTY: MI2-02M

Zařazení :

Ovládací klávesnice pro INELS / sběrnice CIB

Typ :

MI2-02M

EAN kód :

8595188136624

Popis :

Externí master sběrnice CIB

Cena jednotky :

3850

Provedení :

DIN - na lištu

Počet obsazených DIN :

1

Napájecí napětí z CIB :

27

V DC

Napájecí napětí z baterie :

24

V DC

Napájecí napětí ze sítě :

V AC

Jmenovitý proud z CIB :

25

mA (při plném zatížení)

Výstupní napájení sběrnice CIB :

V DC

Výstupní jmenovitý proud do CIB :

mA

Počet sběrnic CIB :

(které může obsloužit)

Spínač :

Stmivač :

Časovač :

Rolety :

Topení :

Počet teplotních čidel :

Počet tlačítek :

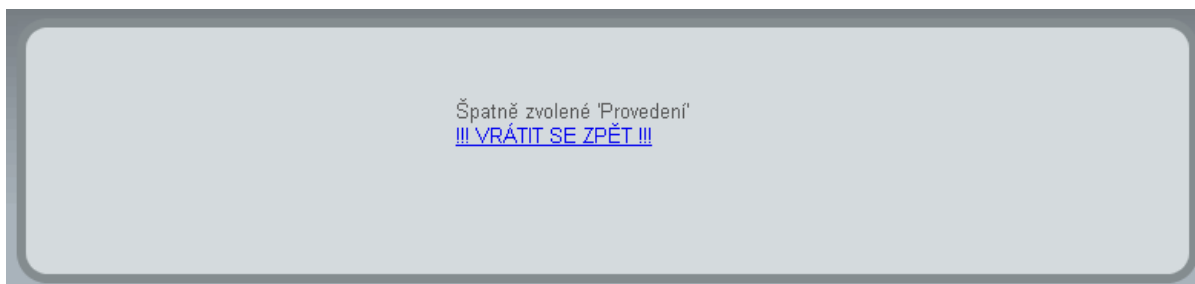
Zpět

Vymazat

Upravit komponentu

Obrázek 6: Prohlížení komponent

Podobný vzhled jak můžeme vidět na obrázku 6 má i možnost „Vytvořit komponentu“. Pokud uživatel zadá špatnou hodnotu některého pole, objeví se chyba (obrázek 7), kde se mu vypíše popis chyby a možnost „Vrátit se zpět“, aby svůj omyl mohl napravit. Je zde především hlídána možnost „Typ“, která musí být zadána. Ostatní možnosti, kromě tedy typu, EAN kódu, popisu a provedení, musejí obsahovat pouze číselné hodnoty. Kolonka „Počet obsazených DIN“ může být zadána pouze v případě, že je v políčku „Provedení“ vybrána možnost „DIN – na lištu“.



Obrázek 7: Chyba s popisem

V rozhraní se taktéž, kromě kontroly čísel a voleb, i kontrola emailových adres a to v kolonce „Úprava správců“ při jejich vytváření a editaci.

5.3.2 Konfigurátor

Konfigurátor se ukrývá pod poslední možností položek z menu s patřičným názvem „Konfigurátor“. Tento systém jsem celkově rozdělil do čtyř kroků pro lepší přehlednost a oddělenost různých typů voleb od sebe.

V prvním kroku, jak vidíme na obrázku 8, je možnost výběru typu budovy a výběru jednotlivých podlaží. V první řadě musí uživatel vybrat typ budovy, až na základě toho výběru se mu zpřístupní výběr jednotlivých pater objektu. V systému jsem ošetřil možnost výběru vyšších pater. K tomu, aby uživatel mohl vybrat druhé, nebo vyšší patro, je potřeba nejprve mít označenou volbu patra nižšího.

Po výběru typu budovy a zvolení příslušných podlaží objektu se uživatel může přesunout k dalšímu kroku konfigurace a to pomocí tlačítka „Další krok“.

V druhém kroku je uživateli nabídnuta volba výběru jednotlivých pokojů v daném podlaží (obrázek 9). Uživatel si zde označí jednotlivé pokoje v příslušných podlažích objektu. U každého podlaží má uživatel možnost vepsání názvu pokoje. Tím se mu zpřístupní patřičné kolonky pro vepsání číselných hodnot. Jiné jak číselné hodnoty v tomto případě nebudou akceptovány. Jsou zde připravené pole ve sloupcích a každý sloupec představuje jiný typ funkce, jak je vidět, těchto funkcí je zde pět.

První funkce „časovač“ je funkce, při které chce uživatel ovládat určité zařízení s časovými prodlevami. Příkladem může být zhasnutí světel po 3 minutách nebo jiné podobné příklady. Funkce „spínač“ je klasickou funkcí dobře známou každému z nás. Jedná se prakticky o sepnutí určitého RLC obvodu, např. ovládání žárovky.

Krok 1 / 4

<input type="radio"/> Obytná budova	<input type="radio"/> Kancelářská budova	<input type="radio"/> Výrobní budova
<input type="checkbox"/> Sklep	<input type="checkbox"/> Přizemí	<input type="checkbox"/> Patro prodejců
<input type="checkbox"/> Přizemí	<input type="checkbox"/> 1.patro	<input type="checkbox"/> Patro kanceláří
<input type="checkbox"/> 1.patro	<input type="checkbox"/> 2.patro	<input type="checkbox"/> Skladovací patro
<input type="checkbox"/> 2.patro	<input type="checkbox"/> 3.patro	<input type="checkbox"/> 1.patro
<input type="checkbox"/> 3.patro	<input type="checkbox"/> 4.patro	<input type="checkbox"/> 2.patro
<input type="checkbox"/> Podkrovní	<input type="checkbox"/> Podkrovní	<input type="checkbox"/> 3.patro

Obrázek 8: Konfigurator – výběr budovy a podlaží

Krok 2 / 4

OBYTNÁ BUDOVA

SKLEP	časovač	spínač	stmívač	rolety	topení
<input checked="" type="checkbox"/> Kotelna	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>	<input type="text" value="0"/>
<input type="checkbox"/> Spíž	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Sklad	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Hala	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Kuchyně	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Obývací pokoj/Jidelna	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Koupelna/WC	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Ložnice	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Pracovna	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> Dětský pokoj	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="checkbox"/> <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Obrázek 9: Konfigurator - volba pokojů a funkcí

Třetí funkcí s názvem „stmívač“. Tato funkce se především využívá pro stmívání světelných zdrojů RLC. To znamená, že světla se zhasínají nebo rozsvěcují postupně v nějakém časovém intervalu a ne jak u spínacích obvodů, kde se tento děj děje „skokově“. Mezi poslední dvě funkce

patří „rolety“ a „topení“. Funkce „rolety“ slouží k tomu, pokud chce uživatel ovládat různé pohony rolety, žaluzií, markýz nebo garážových vrat. Poslední funkce „topení“ slouží uživatelům, kteří chtějí ovládat termohlavice radiátorů nebo jiná topná média.

Po navolení počtu jednotlivých funkcí v daném pokoji se uživatel opět přesune pomocí tlačítka „Další krok“ k dalším volbám (obrázek 10).

Krok 3 / 4

DOPLŇKOVÉ JEDNOTKY:

GSM komunikátor:	<input type="radio"/> ANO <input checked="" type="radio"/> NE
Zabezpečovací klávesnice:	<input type="radio"/> ANO <input checked="" type="radio"/> NE
Záloha systému baterii:	<input type="radio"/> ANO <input checked="" type="radio"/> NE
Operátorský panel - Touch 51/PCB:	<input type="radio"/> ANO <input checked="" type="radio"/> NE

SPÍNAČ:

Typ montáže:: ☒ Instalační krabice ☐ Rozvaděč

ČASOVAČ:

Provedení: ☒ ELEGANT ☐ LOGUS90

OVLÁDÁNÍ STMÍVÁNÍ:

Typ montáže: ☒ Instalační krabice ☐ Rozvaděč

Stmívat zářivky: ☐ ANO ☒ NE

OVLÁDÁNÍ ROLET:

Napětí pro ovládání roletových pohonů:

OVLÁDÁNÍ TOPENÍ:

Napětí pro ovládání termohlavic :

Přejete si i termohlavice? ☐ ANO ☒ NE

Obrázek 10: Konfigurátor - volby systému

V třetím kroku mého systému jsem zvolil možnost uživateli nabídnout různé konfigurace systému. Jednou z těchto možností jsou „doplňkové jednotky“, je zde uvedeno pár jednotek k vylepšení systému. Jednou z možností je GSM komunikátor, který slouží pro komunikaci se systémem přes krátké textové zprávy SMS.

Krok 4 / 4

SYSTÉMOVÉ JEDNOTKY:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
CU2-01M	1	10990,-	10990,-
PS-100/INELS	1	1280,-	1280,-
BPS2-02M	1	1320,-	1320,-

DOPLŇKOVÉ JEDNOTKY:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
GSM2-01	1	9800,-	9800,-
KEY2-01	1	2740,-	2740,-
Touch 51/PCB	1	33290,-	33290,-

SPÍNAČ:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
SA2-04M	2	3080,-	6160,-

ČASOVAČ:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
WSB2-80/E	1	1716,-	1716,-

OVLÁDÁNÍ STMÍVÁNÍ:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
LBC2-02M	3	3295,-	9885,-

OVLÁDÁNÍ ROLET:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
JA2-02B/AC	5	1595,-	7975,-

OVLÁDÁNÍ TOPENÍ:

TYP	POČET	CENA ZA KS	CENA ZA JEDNOTKY
HC2-01B/AC	5	1390,-	6950,-
Alpha 230	5	645,-	3225,-

CELKOVÁ CENA (bez DPH): 95331,-

Zpět

Tisk

Obrázek 11: Konfigurator – výčet jednotek

Možnost zálohy baterie je z jednou z důležitých částí, její vybrání přímo ovlivňuje následný výčet jednotek. Aplikace na základě tohoto výběru vloží do systému jiný typ oddělovače a přepočítá následné doplnění zbývajících systémových jednotek. Ostatní možnosti volby se zpřístupní pouze na základě dříve vložených dat do patřičných polí (Krok 2/4).

U spínače se rozlišuje možnost montáže, je zde k výběru buď instalace do instalační krabice (BOX) a nebo možnost montáže do rozvaděče (DIN). U časovače je k volbě typ provedení, v provedení ELEGANT je počítáno s více možnostmi jako při výběru LOGUS90. Nicméně tento typ provedení má pouze vzhledovou složku a na instalaci nemá vliv jako např. u spínače. Časovací jednotky se montují pouze do instalační krabice a proto zde není možnost výběru montáže. U ovládání stmívání jsou dvě možnosti, jednak je to možnost montáže a navíc je zde možnost jestli chceme stmívat zářivky (tato možnost je ale přístupná pouze při označení typu montáže „rozvaděč“). U ovládání rolet je k výběru výše napětí, uživatel si může vybrat jaké bude dodávané napětí pro pohony těchto jednotek. Stejnou možnost má i u ovládání topení. Navíc jsem zde přidal možnost, jestli si uživatel přeje do výpočtu zahrnout i termohlavice k těmto jednotkám.

Po navolení svých požadavků stačí zmáčknout tlačítko „Vyhodnotit“, tím dojde k propočítání a vyhodnocení všech zadaných hodnot a zobrazí se stránka s výčtem jednotlivých systémových jednotek (obrázek 11).

Na tomto výčtu jasně vidí systémové jednotky, které jsou nejdůležitější pro chod celého systému. Dále zde vidí doplňkové jednotky, které si sám zvolil. A pak už tu je výčet potřebných jednotek. Ten obsahuje typ jednotky, počet jednotek, které budou potřeba a jejich vykalkulování ceny. Pod celým výčtem je celková cena za celý návrh a pochopitelně i možnost tisku této sestavy.

6 Testování

K testování této aplikace jsem především využil rodinné příslušníky všech věkových kategorií. Byl to asi nejlepší způsob jak získat potřebná data vzhledem k tomu, že aplikace by měla sloužit koncovým uživatelům, u kterých musíme počítat i s velkým věkovým rozdílem a různou počítačovou gramotností. Proto jsem k testování přizval i nejstarší členy rodiny, kteří mají pouze mizivé počítačové znalosti. Celkem se testu zúčastnilo 13 členů.

Všichni testovaní účastníci měli za úkol vytvořit libovolný návrh systému podle vlastního uvážení. Před testováním jsem jim vysvětlil, k čemu tento systém mohou využít a co všechno umí. A k jakému účelu je vlastně zkonstruovaný. Dále jsem je jen pozoroval a zjišťoval, zda-li nemají s něčím problém, případně si tento problém poznamenal. Popřípadě zapsal jestli našli řešení v nápovědě nebo ne.

V prvním kroku konfigurator neudělal žádný ze členů chybu a každému se podařilo navolit typ budovy a vybrat si požadovaný počet podlaží objektu.

Při druhém kroku konfiguratoru se již naskytla chyba, dva členové chybně zadávali hodnoty do patričních polí. Zřejmě byli v domněnku, že do připravených polí patří hodnoty „ano“ a „ne“, tudíž místo číselných hodnot zadávali řetězce, což bohužel vedlo k velké chybě na kterou nebyli upozorněni. Ostatní členové svůj úkol zvládli správně, i když čtyři z nich si při tomto kroku přečetli rady v nápovědě.

Krok 3 nedělal nikomu z testovaných žádné problémy. Jen tedy zprvu nevěděli co znamená u „časovače“ provedení ELEGANT nebo LOGUS90. U testovaných subjektů ženského pohlaví se naskytl menší zádrhel, když některé nevěděly, co je to instalační krabice či rozvaděč. Ale žádná vážná chyba zde nenastala. Je to i celkem logické, vzhledem k tomu, že možnosti volby se skládají z jednotek typu „radio button“ a „výběrové pole“. Tudíž určitá hodnota u nich je již standardně dána a je tedy s čím kalkulovat.

V posledním kroku aplikace se již žádná chyba naskytnout nemůže, tudíž jediným úkolem zde bylo vytisknout si zvolenou konfiguraci, což se podařilo všem testovaným subjektům.

V poslední části testu jsem se zúčastněných ptal na jejich osobní názor na tento konfigurační nástroj. Ve většině případů byli ohlasy kladné, což mě velmi potěšilo, i když tato skutečnost je jistě zkreslená rodinnými vztahy. Nejvíce si pochvalovali jednoduchost aplikace a její přehlednost. Také ocenili, že výsledný rozpis měli velmi brzo vyhotovený a byl přehledný.

Mezi připomínkami k systému byli především komentáře k třetímu kroku konfiguratoru, kde by testovaní uvítali více možností volby. Další výtku měla možnost tisku, kdy by někteří uvítali i možnost vygenerování souboru PDF. A v poslední řadě zaznělo, především tedy z mužských řad, že by uvítali kalkulaci vedlejších nákladů, jako délka jednotlivých kabelů, množství instalačních krabic a rozvaděčů.

7 Závěr

V mé práci jsem se zabýval návrhem a realizací webového konfiguračního systému pro systém INELS. Byla zde hlavní snaha především o vyhovění požadavků firmě ELKO EP s.r.o. a vhodné navržení dané aplikace. Dle mého názoru je již aplikace i v tomto stavu vhodným nástrojem pro konfigurace a jistě ulehčí práci a čas při návrhu jednotlivých typů domů.

Mezi výhody lze určitě zmínit snadnost přístupu k této aplikaci. Vzhledem k tomu, že se jedná o webovou konfigurační nástroj, nebude problém k přístupu pro uživatele s různými operačními systémy. Navíc odpadne nutnost jakékoliv instalace, což uživatelé jistě ocení. Jediná podmínka je zde přístup k internetu, což v dnešní době má u nás již bezmála třetina domácností a celosvětově je to přes miliardu uživatelů.

Při testování aplikace jsem zjistil, že je aplikace použitelná i pro široké věkové spektrum uživatelů. A pokud daný člověk zvládá aspoň základní počítačovou gramotnost, tak nebude mít jistě se spuštěním aplikace přílišný problém. Stejně tak i k dobrání určitého výsledku mu nebude dělat přílišné starosti. Teda pokud ovšem nezasílá špatné hodnoty do daných polí. I když je nutno dodat, že se jedná o základní aplikaci, která nenabízí možnost k přílišným chybám.

V dalším rozvoji by se dalo aplikaci rozšířit o mnoho různých nových funkcí. Jistě by bylo vhodné vylepšit tuto aplikaci o návrh vytěžování jednotek nebo o konfigurator designu jednotlivých jednotek. Možnost by též byla ve vytvoření instalovatelné verze této aplikace, kterou by uživatel mohl využívat i bez přístupu k internetu a po připojení sítě by se aktualizoval obsah jednotlivých dat aplikace. Nebo vytvořit zásuvný modul, který by byl součástí již vytvořené aplikace Inels Design & Manager, která slouží pro konfiguraci systému INELS.

Jistě nejzajímavější variantou by se jevila varianta více inteligentní aplikace s vlastním návrhem grafické podoby domu. Uživatel by v jednoduchosti nakreslil typ budovy, nebo spíše naklikal pomocí předpřipravených grafických prvků, vložil patřičné pokoje a pak jen volil co si v jakých místnostech přeje za funkce. Stanovil by určité „pevné body“ (zásuvky, vypínače, osvětlení apod.) v jednotlivých místnostech a aplikace by již sama navrhla nejvhodnější řešení. Dbalo by se na nejvhodnější trasy k vedení sběrnice a silového vedení s ohledem na maximální vytížení, přehlednost a případné další možné rozšíření.

Literatura

- [1] Wikipedia: Inteligence [online]. [cit. 2009-12-31]. Dostupné na URL:
< <http://cs.wikipedia.org/wiki/Inteligence> >
- [2] Wikipedia: Elektroinstalace [online]. [cit. 2009-12-31]. Dostupné na URL:
< <http://cs.wikipedia.org/wiki/Elektroinstalace> >
- [3] INELIN: Popis [online]. [cit. 2009-12-31]. Dostupné na URL:
< <http://www.inelin.cz/popis> >
- [4] Elekrika: Odborníci doporučují KNX [online]. Tisková zpráva z 16.7.2007 [cit. 2009-12-31].
Dostupné na URL: < <http://elektrika.cz/data/clanky/odbornici-doporucuji-knx/view> >
- [5] INELS: obrázek topologie [online]. [staženo 2010-01-01]. Dostupné na URL:
< http://88.81.90.93/media/img/topologie_inels.jpg >
- [6] INELS: O systému [online]. [cit. 2010-01-01]. Dostupné na URL:
< <http://inels.cz/index.php?sekce=view> >
- [7] INELS: Přínosy pro instalatéra [online]. [cit. 2010-01-01]. Dostupné na URL:
< <http://inels.cz/index.php?sekce=view&id=99> >
- [8] Vaniš Jan: Systémová technika budov a bytů [online]. Leden 2003 [cit. 2010-01-02]
Dostupné na URL:
< <http://fei1.vsb.cz/kat452/VSB.CZ/TZB/systemova%20technika%20budov.pdf> >
- [9] ELKO EP s.r.o.: INELS - Technický katalog 2010. ELKO EP, Holešov 2009. www.elkoep.cz
- [10] Zemčík Pavel: Tvorba uživatelských rozhraní [online]. Listopad 2006 [cit. 2010-01-02].
Dostupné na URL: < <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/ITU-IT/texts/> >
- [11] Utěkal Jan: Tvorba uživatelských rozhraní [online]. Prosinec 2009 [cit. 2010-01-02].
Dostupné na URL: < <https://wis.fit.vutbr.cz/FIT/st/course-files-st.php/course/ITU-IT/other/> >
- [12] Wikipedia: PHP [online]. [cit. 2010-01-15]. Dostupné na URL:
< <http://en.wikipedia.org/wiki/PHP> >
- [13] Pankrác Miroslav: *PHP a MySQL bez předchozích znalostí*. Praha, Computer Press, a. s., 2007.
Vydání první, ISBN 978-80-251-1758-3
- [14] Musciano Chudl & Kennedy Bill: *HTML a XHTML kompletní průvodce*. Praha, Computer Press, a. s., 2000. ISBN 80-7226-407-9
- [15] Wikipedia: HTML [online]. [cit. 2010-04-27]. Dostupné na URL:
< <http://en.wikipedia.org/wiki/HTML> >
- [16] Wikipedia: IETF [online]. [cit. 2010-04-27]. Dostupné na URL:
< <http://cs.wikipedia.org/wiki/IETF> >
- [17] Groff James R. & Weinberg Paul N.: *SQL kompletní průvodce*. Brno, CP Books, a. s., 2005.
ISBN 80-251-0369-2

- [18] Interval: Databáze a jazyk SQL [online]. [cit. 2010-04-27]. Dostupné na URL:
< <http://interval.cz/clanky/databaze-a-jazyk-sql/> >
- [19] Živě: Víte co je SQL? Ne? Nevadí – dnes začínáme! [online] [cit. 2010-04-27]. na URL:
< <http://www.zive.cz/clanky/vite-co-je-sql-ne-nevadi---dnes-zaciname/sc-3-a-4320/default.aspx> >
- [20] Wikipedia: SQL [online]. [cit. 2010-04-27]. Dostupné na URL:
< <http://en.wikipedia.org/wiki/Sql> >
- [21] Wikipedia: Kaskádové styly [online]. [cit. 2010-05-02]. Dostupné na URL:
< http://cs.wikipedia.org/wiki/Kask%C3%A1dov%C3%A9_styly >
- [22] Burget Radek: Kaskádové styly (CSS) – Jemný úvod [online]. 2010 [cit. 2010-05-03].
Dostupné na URL: < http://www.fit.vutbr.cz/~burgetr/tws/prednasky/itw_p03.pdf >
- [23] Webtvorba: PHP – Začínáme [online]. 2004 [cit. 2010-05-03]. Dostupné na URL:
< <http://www.webtvorba.cz/php/zaciname.html> >
- [24] Wikipedia: JavaScript [online]. [cit. 2010-05-03]. Dostupné na URL:
< <http://cs.wikipedia.org/wiki/Javascript> >
- [25] Tvorba-webu: JavaScript [online]. [cit. 2010-05-03]. Dostupné na URL:
< <http://www.tvorba-webu.cz/javascript/> >
- [26] Jak psát web: JavaScript [online]. [cit. 2010-05-03]. Dostupné na URL:
< <http://www.jakpsatweb.cz/javascript/> >
- [27] Meyer Eric A.: *CSS – kompletní průvodce*. Brno, ZONER software, s. r. o., 2007.
ISBN 978-80-86815-64-0
- [28] Flanagan David: *JavaScript – kompletní průvodce*. Praha, Computer Press, 2002.
ISBN 80-7226-626-8

Seznam příloh

Příloha 1. Obsah a struktura DVD nosiče 1 stránka

Příloha 2. DVD